



Comparative Study of AI Techniques for Personalized Learning

Assistant Professor Smita Mangesh Junnarkar¹, Karuna Tripathi²

Department of Data Science, SIES COLLEGE OF ARTS, SCIENCE AND COMMERCE (EMPOWERED AUTONOMOUS), Sion (W), Mumbai. *Smita Mangesh Junnarkar*, smitaj@sies.edu.in

Department of Data Science, SIES COLLEGE OF ARTS, SCIENCE AND COMMERCE (EMPOWERED AUTONOMOUS), Sion (W), Mumbai. *Karuna Tripathi*, simplekaruna2006@gmail.com

ABSTRACT—

Learners differ in prior knowledge, pace, goals, and motivation, and a single sequence of lessons seldom suits all of them. Artificial intelligence offers many ways to adapt instruction, but the techniques rest on different assumptions, need different data, and fail in different ways. This paper compares eight families of AI technique used for personalized learning: rule-based expert tutors, Bayesian knowledge tracing, item response theory and cognitive diagnosis, collaborative filtering and matrix factorisation, deep knowledge tracing, reinforcement learning and bandit methods, knowledge-graph and content-based methods, and large language model tutors. For each family the paper explains what the technique models, what data it needs, what it does well, and where it breaks down. A comparison framework is then proposed, with a technique inventory, formal models of mastery, item difficulty, and learning gain, and metrics for prediction, recommendation, learning outcome, interpretability, cost, and fairness. The paper closes with design guidelines and ethical considerations. The work is analytical and reports no new empirical results. Its purpose is to help students, researchers, and developers choose and combine techniques that fit their learners, data, and constraints.

Keywords—*Personalized learning, adaptive learning, intelligent tutoring systems, knowledge tracing, recommender systems, reinforcement learning, large language models, educational data mining, learner modelling*

1. INTRODUCTION

A teacher with thirty students cannot pace a lesson for each of them. Some arrive with gaps from earlier years, some already know the topic, and some need the same idea explained three different ways. Bloom reported that students who received one-to-one tutoring performed about two standard deviations better than students taught in conventional classes, and he asked whether group instruction could be made nearly as effective [1]. Since human tutors cannot be provided to every learner, much of the research on educational technology has tried to supply part of that benefit through software that adapts to the individual. This is what is meant here by personalized learning: adjusting the content, sequence, pace, and feedback of instruction to what each learner knows, wants, and needs.

The evidence that such software can help is encouraging but uneven. A review by VanLehn found that intelligent tutoring systems produced learning gains close to those of human tutors, with effect sizes of roughly 0.76 and 0.79 respectively [2], and a meta-analysis of controlled evaluations by Kulik and Fletcher likewise reported positive effects [3]. A large field trial of a widely used algebra tutor, however, found gains that were modest and that appeared mainly in the second year of use [4]. Results therefore depend on the technique, the subject, the quality of implementation, and the way the system is used in the classroom.

The difficulty for a student, researcher, or developer is that the toolbox is large and its parts come from different traditions. Rule-based tutors grew out of cognitive science, item response theory out of psychometrics, collaborative filtering out of recommender systems, and deep knowledge tracing and large language models out of machine learning. Each community reports its own metrics on its own datasets, so a reader who asks which technique to use finds no common yardstick. Existing reviews catalogue the many applications of artificial intelligence in education [5], [6], [7]; this paper concentrates instead on a side-by-side comparison of the techniques themselves, asking the same questions of each.

The paper addresses three questions. First, which AI techniques are used to personalize learning, and what does each one actually model? Second, how do they compare in terms of data requirements, predictive and pedagogical value, interpretability, cost, and fairness? Third, how should techniques be combined and safeguarded when they are placed in front of real learners? The answers are developed through a description of the personalization pipeline, a classification of eight technique families, a comparison framework, and a set of design guidelines.

The paper is organised as follows. Section 2 summarises earlier studies, and Section 3 explains how personalization works from learner data to adapted instruction. Section 4 describes the eight technique families. Section 5 presents the comparison framework, Section 6 explains how the comparison should be interpreted, and Section 7 lists design guidelines and safeguards. Section 8 discusses limitations and ethical issues, Section 9 suggests future directions, and Section 10 concludes.

2. BACKGROUND AND RELATED STUDIES

2.1 *Origins of the Idea*

Adaptive instruction has a long history. Early computer tutors encoded an expert's knowledge of a subject as rules and compared the learner's steps against them. Corbett and Anderson introduced knowledge tracing, a probabilistic way to estimate from a learner's answers whether a skill had been mastered, and used it to decide when a tutor should stop giving practice on that skill [8]. In parallel, the field of educational data mining grew up around the idea that the logs produced by learning platforms could be analysed to understand and improve learning [9], [10]. Together these lines of work supplied the first generation of learner models, and most later techniques can be read as attempts to improve one of their parts.

2.2 *Evidence on Effectiveness*

The effectiveness studies cited above show that adaptive tutoring can approach human tutoring in well-structured domains [2], [3], and that results in schools at scale are more modest [4]. Comparisons between techniques are rarer and harder to interpret, because predictive accuracy on a dataset is not the same thing as learning gain in a classroom. Gervet and colleagues compared several knowledge tracing approaches on multiple datasets and found that the best choice depended on dataset size and on how much the order of interactions mattered, with simpler logistic models remaining competitive on smaller collections [11]. The lesson for the present study is that a more complex technique is not automatically a better one.

2.3 *The Knowledge Tracing Lineage*

Bayesian knowledge tracing was later extended to give each learner individual parameters [12]. Piech et al. replaced hand-built probabilistic models with a recurrent neural network built from long short-term memory units [13] and called the result deep knowledge tracing [14]. Memory-augmented networks [15] and attention-based models derived from the transformer architecture [16] followed, including self-attentive [17] and context-aware [18] variants. A recent survey organises this growing family and its applications [19]. The progression shows a steady shift from models designed by experts to models learned from data.

2.4 *Recommendation, Decision Making, Language Models, and Risks*

Recommender systems offered a second route to personalization. Matrix factorisation [20] and neural collaborative filtering [21] predict what a user will like from the behaviour of similar users, and the same ideas have been applied to suggesting exercises and resources [22]. Reinforcement learning [23], [24] treats instruction as a sequence of decisions, and multi-armed bandits offer a simpler form suited to choosing among activities [25]. Most recently, large language models [26] have been proposed as tutors, writers of explanations, and conversational partners [27]. Alongside these advances, concerns have grown about hallucinated content [28], about the limits of models trained on large text collections [29], and about bias in educational algorithms [30]. India's Digital Personal Data Protection Act, 2023 places duties on those who decide why and how personal data are processed [31], which matters because every technique discussed here depends on learner data. For this reason the present paper treats legal compliance as a floor and not as a complete answer.

3. HOW PERSONALIZATION WORKS

To compare techniques fairly, it helps to see where each one sits in a common pipeline. There are five steps: sensing, learner modelling, content modelling, adaptation, and feedback. Different techniques occupy different steps, and several of them can be combined within one system.

3.1 *Sensing*

Every interaction with a learning platform produces events: an answer, the time taken to give it, a hint requested, a video paused, a message typed. Most events are logged automatically with a timestamp, a learner identifier, and the item involved. The quality of every later step depends on what is recorded here, and a technique that needs rich logs cannot be used on a platform that stores only final scores.

3.2 *Learner Modelling*

The learner model turns raw events into an estimate of the learner's state, such as the probability that each skill is mastered, a single ability score, a vector of latent features, or a set of preferences. Bayesian knowledge tracing, item response theory, and deep knowledge tracing all work mainly at this step. The model is usually hidden from the learner, yet it determines almost everything the system later does.

3.3 *Content Modelling*

Adaptation also requires a description of what is being learned: which skills exist, which items test them, how hard each item is, and which skills depend on others. This description may be written by experts, estimated from data, or stored as a knowledge graph [32]. A learner model that knows a student is weak in fractions is of little use unless the content model says which exercises address fractions and what should come before them.

3.4 Adaptation

The adaptation step chooses what happens next: the next item, a hint, a worked example, a review of an earlier topic, or a different path through the course. Rules, recommenders, bandits, and reinforcement learning policies operate here, and language models can generate the explanation or question itself. This is the step the learner actually experiences.

3.5 Feedback and Evaluation

The learner's response to the adapted material becomes new data, so the system does not merely observe learning but shapes the data it will later be trained on. If a recommender only shows easy items, it will only collect evidence about easy items. Evaluation must therefore look beyond prediction accuracy to whether the learner learned more, and it must be repeated as the system and its users change.

4. AI TECHNIQUES COMPARED

The techniques below are sorted into eight families. For each one the discussion covers what it models, what data it needs, what it does well, and where it breaks down. The reasoning rests on published studies and on how such systems are commonly built. It is meant to guide comparison, not to claim that every product uses every technique in the form described.

4.1 Rule-Based and Expert-Model Tutors

In a rule-based tutor, experts write down how to solve problems in a domain as production rules or constraints, and the system follows the learner's steps against that model, giving hints when a step goes wrong. The learner model is usually a simple record of which rules the learner has applied correctly. Tutors of this kind account for much of the positive evidence reported for intelligent tutoring [2], [3], [4]. Their strengths are transparency and tight pedagogical control, since every hint can be traced to a rule. Their weakness is authoring cost: building the expert model takes considerable effort, and the result is confined to the narrow domain it was written for. Adaptation is mostly branching rather than learning from data.

4.2 Bayesian Knowledge Tracing

Bayesian knowledge tracing treats each skill as a hidden state that is either known or unknown, and updates the probability of mastery after every answer. Four parameters describe a skill: the prior probability that it is already known, the probability of learning it at each opportunity, the probability of guessing correctly without knowing it, and the probability of slipping despite knowing it [8]. Individualised versions fit parameters per learner [12]. The model is easy to interpret and needs little data per learner, and its output, a probability of mastery, can directly drive mastery learning, where practice continues until a threshold is passed. Its limits follow from its assumptions: skills are treated as independent, forgetting is ignored, and knowledge is binary, so it cannot capture partial understanding or relationships between topics.

4.3 Item Response Theory and Cognitive Diagnosis

Item response theory comes from psychometrics. It models the probability that a learner answers an item correctly as a function of a latent ability and of item properties such as difficulty, discrimination, and the chance of guessing [33]. Because items are calibrated on a population, ability estimates are comparable across learners and tests, and the model supports computerised adaptive testing, in which the next item is chosen to be maximally informative about the learner's ability. Cognitive diagnosis models extend the idea by linking each item to the skills it requires, so that the output is a profile of skills and not a single score. These methods work with modest data and have a strong statistical basis. Their main limitation is that ability is treated as fixed within a session, so they describe where a learner stands better than they describe how the learner is changing.

4.4 Collaborative Filtering and Matrix Factorisation

Collaborative filtering recommends material that similar learners found useful. Matrix factorisation represents learners and resources as vectors in a shared space and scores their match [20], and neural variants learn richer interactions [21]. The approach is attractive for large catalogues of videos, articles, and exercises, where no expert could assign each item by hand [22]. It has three characteristic weaknesses. It suffers from cold start, since new learners and new items have no history. It tends to favour popular items. And it optimises what learners click or rate, which is not the same as what they learn, and it knows nothing of prerequisites unless they are added from elsewhere.

4.5 Deep Knowledge Tracing

Deep knowledge tracing predicts the learner's next answer from the whole sequence of earlier ones. Recurrent networks [14], memory networks [15], and attention-based models [17], [18] learn the representation of the learner's state from data, so skills do not have to be tagged and relations between them can emerge. Graph neural networks [34] can be added to bring in prerequisite structure. These models can use rich inputs such as response time and item text, and they often reach high predictive accuracy on large datasets. They also need such datasets, they are difficult to interpret, and their predictions can be less stable than those of simpler models. The comparison by Gervet et al. suggests that the advantage over logistic models depends on how much data and how long a sequence are available [11].

4.6 Reinforcement Learning and Bandit Methods

Reinforcement learning chooses actions so as to maximise a long-term reward [23]. In a learning setting, the action may be the next activity, the state is the learner model, and the reward is some measure of learning. Deep reinforcement learning [24] allows large state spaces. Multi-armed bandits are a simpler form, in which the system balances trying uncertain activities against repeating those known to work, and they have been applied to choosing among activities in tutoring systems [25]. The main attraction is that the system optimises the sequence of instruction and not just the next prediction. The main difficulty, noted in a review of the field, is that defining the reward and evaluating a policy without harming real learners are hard problems [35]. Learning also needs many interactions, so policies are often trained in simulation, where the simulated learner may differ from real ones.

4.7 Knowledge Graph and Content-Based Methods

A knowledge graph stores skills, concepts, and resources as nodes and their relations, such as prerequisites, as edges [32]. Personalization then follows the graph: a learner who has not mastered a prerequisite is steered back to it, and resources can be matched to learners by their content. The great advantage is explainability, because a learning path can be justified in terms the teacher recognises. Content-based methods also avoid the cold-start problem for items, since a new resource can be placed by its description. The costs are construction and maintenance of the graph, and the fact that a graph describes the subject and not the learner. It therefore needs a learner model beside it.

4.8 Large Language Model Tutors

Large language models [26] can generate explanations at different levels, create practice questions, give feedback on written answers, and hold a conversation with the learner [27]. This flexibility is new: earlier techniques chose among prepared content, whereas a language model can produce content on demand. Retrieval-augmented generation can ground its answers in the course's own material [36]. The weaknesses are equally clear. Models can state falsehoods confidently [28], can reflect the limitations of their training text [29], may be inconsistent in their teaching approach, and carry computing costs and privacy concerns when learner text is sent to external services. A language model has no persistent, calibrated estimate of what a learner knows unless one is supplied by another component.

5. FRAMEWORK FOR COMPARING TECHNIQUES

A fair comparison needs a common yardstick. The framework below is independent of any one dataset or product, so it can be applied to a research study, to a school's procurement decision, or to a developer's design choice.

5.1 Scope and Participants

The first step is to define the learner population, the subject, and the decision that personalization is meant to improve, such as which item to show next or which resource to recommend. For any empirical use, all techniques should be run on the same learner logs with identical splits, and learning outcomes should be measured in a controlled study with a pre-test and a post-test. Informed consent, and a data-handling plan that covers children where relevant, should be in place before anything is collected (see Section 8).

5.2 Technique Inventory

Table 1 lists, for each family, the learner data it needs and what it personalizes. For every technique the assessor should note whether the required data are actually available on the platform, how much expert effort is needed, and whether learner data leave the institution.

Table 1. Learner data required and what each technique personalizes

Technique family	Learner data needed	What it personalizes
Rule-based tutors	Step-level actions and errors	Hints, feedback, and problem selection
Bayesian knowledge tracing	Correct and incorrect answers per skill	Practice amount and mastery decisions
Item response theory	Responses across many learners and items	Item difficulty and adaptive tests
Collaborative filtering	Ratings, clicks, or completions by many users	Resource and course recommendations
Deep knowledge tracing	Long interaction sequences, often with timing	Prediction of next answer and learner state

Reinforcement learning and bandits	Interactions with a measurable outcome	Order and choice of activities
Knowledge graphs	Skill structure and item tags	Learning paths and prerequisite review
Language model tutors	Learner text and course material	Explanations, questions, and dialogue

5.3 Quantifying Learner Knowledge and Learning

Three simple formulas show how the families differ in what they quantify. In Bayesian knowledge tracing, let L_t mean that the skill is known after the t -th opportunity, let s be the slip probability, g the guess probability, and pT the probability of learning at each step. After a correct answer, the updated probability of mastery is as follows.

$$P(L_t | \text{correct}) = P(L_{t-1})(1-s) / [P(L_{t-1})(1-s) + (1-P(L_{t-1}))g]$$

The learner may then learn the skill at this opportunity, so the probability carried forward is $P(L_t) = P(L_t | \text{obs}) + (1 - P(L_t | \text{obs}))pT$, where the first term is the posterior after the observed answer. A skill is usually declared mastered when this value passes a threshold, often 0.95 in practice. Item response theory instead gives the probability that a learner of ability θ answers an item correctly in the three-parameter form.

$$P(X=1 | \theta) = c + (1-c) / (1 + e^{-a(\theta-b)})$$

Here b is the difficulty of the item, a its discrimination, and c the chance of a correct guess. The Rasch model is the special case with $a = 1$ and $c = 0$. Finally, learning outcomes can be compared across courses with different starting points using the normalised gain, where pre and $post$ are percentage scores before and after instruction [37].

$$g = (post - pre) / (100 - pre)$$

For bandit and reinforcement learning methods, performance is often summarised as regret, the cumulative loss against always choosing the best action. If μ^* is the mean reward of the best activity and $\mu_a(t)$ that of the activity chosen at step t , regret over T steps is $RT = \sum (\mu^* - \mu_a(t))$, summed from $t = 1$ to T . These formulas are not directly comparable with one another, which is why the metrics in Section 5.5 are reported side by side and not merged into one score.

5.4 Evaluation Protocol

To compare techniques, they are trained on the same data and evaluated on learners who were not seen in training. Splitting by learner and not by interaction is essential, otherwise the memorising of individuals would be mistaken for genuine prediction. Strong simple baselines, such as logistic regression and Bayesian knowledge tracing, should always be included, because a complex model that cannot beat them has not justified its cost [11]. Cold-start performance should be tested separately, by evaluating on learners and items with little or no history. Comparing several models helps separate the contribution of the data and the technique from the skill of one particular implementation.

5.5 Metrics

Table 2 defines the measures. Predictive accuracy, ranking quality, and regret describe how well a technique estimates or chooses. Learning gain describes whether learners benefit. Interpretability, cost, and fairness describe whether the technique can be used responsibly. Using all of them gives a fuller picture than any single figure.

Table 2. Comparison metrics

Metric	Definition	Purpose
AUC and RMSE	Area under the ROC curve and root mean squared error of predicted correctness on unseen learners	Accuracy of learner modelling
Ranking quality	Normalised discounted cumulative gain of recommended	Quality of recommendations

	resources [38]	
Normalised gain	Share of possible improvement achieved between pre-test and post-test [37]	Actual learning benefit
Cumulative regret	Loss against always choosing the best activity	Efficiency of sequencing
Interpretability	Whether a teacher can inspect why a decision was made	Trust and accountability
Cost	Authoring effort, data needed, computing, and latency	Practical feasibility
Fairness gap	Difference in a chosen metric between learner groups	Equity of outcomes

Every reported value should carry a confidence interval, for example from bootstrap resampling over learners. Because the same learners are used with several techniques, differences between techniques should be checked with a paired test such as the Wilcoxon signed-rank test, which does not assume normally distributed scores.

6. INTERPRETING THE COMPARISON

6.1 Matching Technique to Problem

Three questions narrow the choice. What needs to be predicted or decided: mastery, the next item, a resource, or an explanation? What data exist, and how many learners and interactions are there? How much explanation do teachers and learners need? Table 3 summarises the main strengths, limitations, and best-suited uses of each family, drawing on the descriptions in Section 4.

Table 3. Strengths, limitations, and best-suited uses

Technique family	Main strength	Main limitation	Best suited to
Rule-based tutors	Transparent, pedagogically controlled	High authoring cost, narrow domain	Well-structured subjects such as algebra
Bayesian knowledge tracing	Interpretable, little data needed	Binary skills, no forgetting	Mastery learning with tagged skills
Item response theory	Calibrated, statistically principled	Ability treated as fixed	Adaptive testing and diagnosis
Collaborative filtering	Scales to large catalogues	Cold start, popularity bias	Resource recommendation
Deep knowledge tracing	Learns from sequences, no skill tags	Data hungry, opaque	Large platforms with long logs
Reinforcement learning	Optimises the whole sequence	Reward design, needs many interactions	Sequencing after offline testing

Knowledge graphs	Explainable learning paths	Costly to build and maintain	Subjects with clear prerequisites
Language model tutors	Flexible dialogue and explanation	Errors, cost, privacy	Open-ended help, grounded in course material

6.2 Likely Order of Suitability

The true ranking depends on the setting, yet the mechanisms suggest a general pattern. For a new course with few learners, rule-based methods, Bayesian knowledge tracing, and item response theory tend to rank high because they need little data and are easy to inspect. For a platform with millions of logged interactions, deep knowledge tracing becomes more attractive, although the evidence suggests that it should earn its place against simpler baselines [11]. For recommending resources from a large catalogue, collaborative filtering is natural. For open-ended questions and explanations, language models are the only family that can respond flexibly, provided their output is grounded and checked. Reinforcement learning is best treated as a later stage, once a reliable learner model and a safe way to evaluate policies exist.

6.3 Hybrid Systems

Few real systems rely on a single technique. A natural division of labour is to let a knowledge tracing model estimate the learner's state, a knowledge graph supply prerequisite constraints, a bandit or recommender choose among permitted activities, and a language model write the explanation or hint. In such a design the language model is the interface and not the judge of mastery, which keeps the estimate of what the learner knows in a component that can be calibrated and audited. The comparison framework applies to each component and to the combination, because errors in one part can propagate to the others.

6.4 Cold Start and Data Scarcity

Cold start is the most common practical obstacle. A new learner has no history, a new item has no responses, and a new course has no logs at all. Techniques that rely on expert structure, namely rules, knowledge graphs, and content-based methods, cope best because they do not need past behaviour. Data-driven methods need a bridge: priors from similar learners, a short diagnostic test at the start, or a simpler model that is used until enough data have accumulated. The comparison should therefore always report performance for the first few interactions as well as the average.

6.5 Interpretability, Cost, and Trust

Teachers will use a recommendation only if they can see why it was made, and learners will follow it only if they trust it. Rules, Bayesian knowledge tracing, item response theory, and knowledge graphs offer explanations in terms of skills and prerequisites. Deep models and language models do not, and post-hoc explanations of their output are approximations. Cost has several parts: authoring effort is highest for rules and graphs, data requirements are highest for deep models, and computing and per-query charges are highest for language models. No technique is best on every measure, which is the reason for reporting all of them.

7. DESIGN GUIDELINES AND SAFEGUARDS

7.1 Learners and Teachers

Personalization should support teachers and not replace them. Teachers should be able to see the learner model, understand the reasons for recommendations, and override them. Learners should be given some control, such as the choice to review a topic or skip ahead, because systems that remove all choice risk reducing motivation and responsibility for learning.

7.2 Institutions and Data Governance

The guiding rule is to collect only what the function needs, to state the purpose, and to delete when the purpose ends. These principles are reflected in India's Digital Personal Data Protection Act [31], which also places special duties on the processing of children's data, including the consent of a parent or guardian. Practical steps include a data inventory, short retention for raw logs, access controls with audit trails, and careful review of any external service, including language model providers, to which learner text is sent.

7.3 Technical Methods

Begin with simple baselines and add complexity only when it is shown to help. Evaluate offline before any learner is exposed, and calibrate predicted probabilities so that a stated 80 per cent means roughly 80 per cent. For language models, ground answers in course material [36], restrict them to the

task, and have teachers review generated content before it is reused. Test for unequal performance across groups of learners [30]. Where data must be protected, federated learning keeps raw data on the device and shares only model updates [39], and differential privacy limits what any output reveals about one person [40], although each reduces accuracy or adds complexity.

7.4 Evaluation and Monitoring

A system should be judged on learning outcomes and not only on engagement or prediction accuracy. Controlled comparisons, such as randomised trials or A/B tests with pre-tests and post-tests, give the most reliable evidence. Monitoring should continue after launch, because learner populations, curricula, and the system itself change, and a model that was well calibrated last year may drift.

7.5 Interface and Policy

Learners should be told in plain language that their activity is used to adapt the material, what is recorded, and what the system concluded about them. Dashboards that let people view, correct, and delete their data give practical meaning to their rights. Institutions should require an impact assessment before a new adaptive feature is launched, set limits on automated decisions with high stakes such as grading or placement, and keep a human route for appeal.

8. LIMITATIONS AND ETHICAL ISSUES

The study has limits. It is analytical, so it contains no newly collected measurements, and its statements about the relative performance of techniques rest on published studies whose datasets, subjects, and years differ from those of any particular reader. Many of the cited results come from specific platforms and countries and may not transfer to every setting, including Indian classrooms with shared devices, limited connectivity, and multilingual learners. The field also changes quickly: language model tutors in particular are evolving faster than the evidence about them. The eight families overlap in practice, and real systems blend them. Finally, the paper cannot say how any particular commercial product works, since such details are generally not public.

Several ethical points also deserve attention. Models trained on past learner data may reproduce existing inequalities, for example by recommending easier material to groups that were historically given it [30]. Constant monitoring can make learners feel watched, and data about a child's mistakes can follow them in ways they never intended. Learners who are weakest, youngest, or least connected may benefit least from systems designed around the average user. Generated content can be wrong or biased [28], [29], and a confident but mistaken explanation can mislead a learner who has no way to check it. Anyone collecting data should obtain informed consent, keep records only as long as needed, and store them securely. It would also be wrong to conclude from this analysis that technology can replace teachers, since the evidence reviewed here concerns tools that work within a human setting and not in place of it.

9. FUTURE DIRECTIONS

Several directions follow from this analysis. The most immediate is to run the comparison framework on shared, openly available learner datasets and to publish the protocols so that other groups can compare findings on equal terms. A second is to study hybrid designs in which a calibrated learner model controls what a language model is allowed to say. A third is to examine how techniques perform in the first few interactions, where cold start decides whether a learner stays. A fourth is to examine conditions specific to India, such as multilingual content, low-bandwidth access, and shared devices, in the light of the Digital Personal Data Protection Act. Finally, evaluating federated learning and differential privacy on realistic educational logs would show how much protection survives at useful accuracy, and controlled classroom trials would show which of the techniques actually improve learning and not just prediction.

10. CONCLUSION

Personalized learning is not a single technology but a collection of techniques that answer different questions. This paper compared eight families, namely rule-based tutors, Bayesian knowledge tracing, item response theory, collaborative filtering, deep knowledge tracing, reinforcement learning and bandits, knowledge graphs, and language model tutors. Expert-built and probabilistic methods are transparent and work with little data, deep and recommender methods exploit large logs but are harder to explain and suffer from cold start, reinforcement learning optimises sequences but is difficult to evaluate safely, and language models offer flexibility at the price of reliability and cost. Because each family is strong where another is weak, judging them one at a time, or by predictive accuracy alone, gives an incomplete picture.

The framework proposed here joins a technique inventory, formal models of knowledge and learning gain, an evaluation protocol that splits by learner, and metrics that include interpretability, cost, and fairness, and so offers a structured way to choose among techniques. The design guidelines suggest that the most dependable systems are hybrids that keep an auditable learner model at the centre, use language models for communication under supervision, and are judged on learning outcomes. It is hoped that the paper encourages students and developers to choose techniques for what their learners and data actually need, so that personalization serves the learner and does not merely follow the data.

REFERENCES

- [1]B. S. Bloom, "The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring," *Educational Researcher*, vol. 13, no. 6, pp. 4--16, 1984.
- [2]K. VanLehn, "The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems," *Educational Psychologist*, vol.

46, no. 4, pp. 197--221, 2011.

- [3]J. A. Kulik and J. D. Fletcher, "Effectiveness of intelligent tutoring systems: A meta-analytic review," *Review of Educational Research*, vol. 86, no. 1, pp. 42--78, 2016.
- [4]J. F. Pane, B. A. Griffin, D. F. McCaffrey, and R. Karam, "Effectiveness of Cognitive Tutor Algebra I at scale," *Educational Evaluation and Policy Analysis*, vol. 36, no. 2, pp. 127--144, 2014.
- [5]O. Zawacki-Richter, V. I. Marin, M. Bond, and F. Gouverneur, "Systematic review of research on artificial intelligence applications in higher education -- where are the educators?" *International Journal of Educational Technology in Higher Education*, vol. 16, art. 39, 2019.
- [6]L. Chen, P. Chen, and Z. Lin, "Artificial intelligence in education: A review," *IEEE Access*, vol. 8, pp. 75264--75278, 2020.
- [7]K. Zhang and A. B. Aslan, "AI technologies for education: Recent research and future directions," *Computers and Education: Artificial Intelligence*, vol. 2, art. 100025, 2021.
- [8]A. T. Corbett and J. R. Anderson, "Knowledge tracing: Modeling the acquisition of procedural knowledge," *User Modeling and User-Adapted Interaction*, vol. 4, no. 4, pp. 253--278, 1994.
- [9]R. S. J. d. Baker and P. S. Inventado, "Educational data mining and learning analytics," in *Learning Analytics: From Research to Practice*. New York, NY, USA: Springer, 2014, pp. 61--75.
- [10]C. Romero and S. Ventura, "Educational data mining and learning analytics: An updated survey," *WIREs Data Mining and Knowledge Discovery*, vol. 10, no. 3, art. e1355, 2020.
- [11]T. Gervet, K. Koedinger, J. Schneider, and T. Mitchell, "When is deep learning the best approach to knowledge tracing?" *Journal of Educational Data Mining*, vol. 12, no. 3, pp. 31--54, 2020.
- [12]M. V. Yudelson, K. R. Koedinger, and G. J. Gordon, "Individualized Bayesian knowledge tracing models," in *Artificial Intelligence in Education (AIED)*, Lecture Notes in Computer Science, vol. 7926, 2013, pp. 171--180.
- [13]S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735--1780, 1997.
- [14]C. Piech et al., "Deep knowledge tracing," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 28, 2015, pp. 505--513.
- [15]J. Zhang, X. Shi, I. King, and D.-Y. Yeung, "Dynamic key-value memory networks for knowledge tracing," in *Proc. International World Wide Web Conference (WWW)*, 2017, pp. 765--774.
- [16]A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 30, 2017, pp. 5998--6008.
- [17]S. Pandey and G. Karypis, "A self-attentive model for knowledge tracing," in *Proc. International Conference on Educational Data Mining (EDM)*, 2019, pp. 384--389.
- [18]A. Ghosh, N. Heffernan, and A. S. Lan, "Context-aware attentive knowledge tracing," in *Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD)*, 2020, pp. 2330--2339.
- [19]G. Abdelrahman, Q. Wang, and B. P. Nunes, "Knowledge tracing: A survey," *ACM Computing Surveys*, vol. 55, no. 11, art. 224, 2023.
- [20]Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30--37, 2009.
- [21]X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. International World Wide Web Conference (WWW)*, 2017, pp. 173--182.
- [22]F. Ricci, L. Rokach, and B. Shapira, Eds., *Recommender Systems Handbook*, 2nd ed. New York, NY, USA: Springer, 2015.
- [23]R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.
- [24]V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529--533, 2015.
- [25]B. Clement, D. Roy, P.-Y. Oudeyer, and M. Lopes, "Multi-armed bandits for intelligent tutoring systems," *Journal of Educational Data Mining*, vol. 7, no. 2, pp. 20--48, 2015.
- [26]T. B. Brown et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 1877--1901.
- [27]E. Kasneci et al., "ChatGPT for good? On opportunities and challenges of large language models for education," *Learning and Individual Differences*, vol. 103, art. 102274, 2023.
- [28]Z. Ji et al., "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, art. 248, 2023.
- [29]E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?" in *Proc. ACM Conf. Fairness, Accountability, and Transparency (FAccT)*, 2021, pp. 610--623.
- [30]R. S. Baker and A. Hawn, "Algorithmic bias in education," *International Journal of Artificial Intelligence in Education*, vol. 32, no. 4, pp. 1052--1092, 2022.
- [31]Government of India, "The Digital Personal Data Protection Act, 2023 (Act No. 22 of 2023)," 2023.
- [32]A. Hogan et al., "Knowledge graphs," *ACM Computing Surveys*, vol. 54, no. 4, art. 71, 2021.
- [33]S. E. Embretson and S. P. Reise, *Item Response Theory for Psychologists*. Mahwah, NJ, USA: Lawrence Erlbaum Associates, 2000.
- [34]T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. International Conference on Learning Representations (ICLR)*, 2017.
- [35]S. Doroudi, V. Alevan, and E. Brunskill, "Where's the reward? A review of reinforcement learning for instructional sequencing," *International Journal of Artificial Intelligence in Education*, vol. 29, no. 4, pp. 568--620, 2019.
- [36]P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 9459--9474.
- [37]R. R. Hake, "Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses," *American Journal of Physics*, vol. 66, no. 1, pp. 64--74, 1998.
- [38]K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Transactions on Information Systems*, vol. 20, no. 4, pp. 422--446, 2002.

-
- [39]B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," Proceedings of Machine Learning Research, vol. 54, pp. 1273--1282, 2017.
- [40]C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," Foundations and Trends in Theoretical Computer Science, vol. 9, no. 3--4, pp. 211--407, 2014.